
Algorithms for Linear Bandits on Polyhedral Sets

Manjesh K Hanawal
 Department of ECE
 Boston University
 Boston, MA 02215
 mhanawal@bu.edu

Amir Leshem
 Department of EE
 Bar-Ilan University
 Ramat-Gan, Israel 52900
 leshema@eng.biu.ac.il

Venkatesh Saligrama
 Department of ECE
 Boston University
 Boston, MA 02215
 srv@bu.edu

Abstract

We study stochastic linear optimization problem with bandit feedback. The set of arms take values in an N -dimensional space and belong to a bounded polyhedron described by finitely many linear inequalities. We provide a lower bound for the expected regret that scales as $\Omega(N \log T)$. We then provide a nearly optimal algorithm that alternates between exploration and exploitation intervals and show that its expected regret scales as $O(N \log^{1+\epsilon}(T))$ for an arbitrary small $\epsilon > 0$. We also present an algorithm that achieves the optimal regret when sub-Gaussian parameter of the noise is known. Our key insight is that for a polyhedron the optimal arm is robust to small perturbations in the reward function. Consequently, a greedily selected arm is guaranteed to be optimal when the estimation error falls below some suitable threshold. Our solution resolves a question posed by [1] that left open the possibility of efficient algorithms with asymptotic logarithmic regret bounds. We also show that the regret upper bounds hold with probability 1. Our numerical investigations show that while theoretical results are asymptotic the performance of our algorithms compares favorably to state-of-the-art algorithms in finite time as well.

1 Introduction

Stochastic bandits are sequential decision making problems where a learner plays an action in each round and observes the corresponding reward. The goal of the learner is to collect as much reward as possible or, alternatively minimize regret over a period of T rounds. *Stochastic linear bandits* are a class of *structured bandit problems* where the rewards from different actions are correlated. In particular, the expected reward of each action or arm is expressed as an inner product of a feature vector associated with the action and an unknown parameter which is identical for all the arms. With this structure, one can infer reward of arms that are not yet played from the observed rewards of other arms. This allows for considering cases where number of arms can be unbounded and playing each arm is infeasible.

Stochastic linear bandits have found rich applications in many fields including web advertisements [2], recommendation systems [3], packet routing, revenue management, etc. In many applications the set of actions are often defined by a finite set of constraints. For example, in packet routing, the amount of traffic to be routed on a link is constrained by its capacity. In web-advertisements

problems, the budget constraints determine the set of available advertisements. It follows that the each arm in these applications belongs to a polyhedron.

Bandit algorithms are evaluated by comparing their cumulative reward against the optimal achievable cumulative reward and the difference is referred to as regret. The focus of this paper is on characterizing asymptotic bounds for regret for fixed but unknown reward distributions, which are commonly referred to as problem dependent bounds [4].

We consider linear bandits where the arms take values in an N -dimensional space and belong to a bounded polyhedron described by finitely many linear inequalities. We derive an asymptotic lower bound of $\Omega(N \log T)$ for this problem and present an algorithm that is (almost) asymptotically optimal. Our solution resolves a question posed by [1] that left open the possibility of efficient algorithms with asymptotic logarithmic regret bounds. Our algorithm alternates between exploration and exploitation phases, where a set of arms on the boundary of the polyhedron is played in exploration phases and a greedily selected arm is played super-exponentially many times in the exploitation phase. Due to the simple nature of the strategy we are able to provide upper bounds which hold almost surely. We show that our regret concentrates around its expected value with probability one for all T . In contrast regret for upper confidence bound based algorithms concentrates only at a polynomial rate [5]. Thus, our algorithms are more suitable for risk-averse decision making. A summary of our results and comparison of regrets bounds is given in Table 1. Numerical experiments show that its regret performance compares well against state-of-the-art linear bandit algorithms even for reasonably small rounds while being significantly better asymptotically.

	K -armed bandits		Linear bandits	
	dependent	independent	dependent	independent
Lower bounds	$K \log T$	\sqrt{KT}	$N \log T$	$N\sqrt{T}$
Upper bounds	$K \log T$	\sqrt{KT}	$N \log^{1+\epsilon} T$	$N\sqrt{T}$
Efficient algorithm	UCB1 [6]	MOSS [7]	SEE (this paper)	<i>ConfidenceBall</i> ₂ [4]

Table 1: Summary of (problem) dependent and (problem) independent regret bounds in multi-armed bandits and linear bandits. We considered linear bandits over a bounded subset of N -dimensional subspace with $\Delta > 0$. The column with bold letters presents the bounds obtained in this paper.

Related Work: Our regret bounds are related to those described in [4], who describe an algorithm (*ConfidenceBall*₂) with regret bounds that scale as $O((N^2/\Delta) \log^3 T)$, where Δ is the reward gap defined over extremal points. These algorithms belong to the class of so called OFU algorithms (optimism in the face of uncertainty). Since OFU algorithms play only extremal points (arms), one may think that $\log T$ regret bounds can be attained for linear bandits by treating them as K -armed bandits, where K denotes the number of extremal points of the set of actions. This possibility arises from the classical results on the K -armed bandit problem due to Lai and Robbins [8] who provided a complete characterization of expected regret by establishing a problem dependent lower bound of $\Omega(K \log T)$ and then providing an asymptotically (optimal) algorithm with a matching upper bound. But, as noted in [1][Sec 4.1, Example 4.5], the number of extremal points can be exponential in N , and this renders such adaptation of multi-armed bandits algorithm inefficient. In the same paper, the authors pose it as an open problem to develop efficient algorithms for linear bandits over polyhedral set of arms that have logarithmic regret. They also remark that since convex hull of a polyhedron is not strongly convex, regret guarantees of their PEGE (Phased Exploration Greedy Exploitation) algorithm does not hold.

Our work is close to FEL (Forced Exploration for Linear bandits) algorithm developed in [17]. FEL separates the exploration and exploitation phases by comparing the current round number against a predetermined sequence. FEL plays randomly selected arms in the exploration intervals and greedily selected arms in the exploitation intervals. However, our policy differs from FEL as follows— 1) we always play fixed set of arms (deterministic) in the exploration phases. 2) noise is assumed to be bounded in [17], whereas we consider more general sub-Gaussian noise model 3) unlike FEL, our policy does not require computationally costly matrix inversions. FEL provides expected regret guarantee of only $\mathcal{O}(c \log^2 T)$ whereas our policy PolyLin has optimal $\mathcal{O}(N \log T)$ regret guarantee. Moreover, the authors in [17] remark that the leading constant c in their regret bound can be set proportional to \sqrt{N} (see discussion following Th 2.4 in [17]), but this seems incorrect in light of the lower bound of $\Omega(N \log T)$ we establish in this paper.

In contrast to the asymptotic setting considered here, much of the machine learning literature deals with problem independent bounds. These bounds on regret apply in finite time and for the minimax case, namely, for the worst-case over all reward (probability) distributions. [9] established a problem independent lower bound of $\Omega(\sqrt{KT})$ for multi-armed bandits, and was shown to be achievable in [7]. For linear bandits, problem dependent bounds are well studied and stated in terms of dimension of the set of arms rather than its size. In [10], for the case of finite number of arms, a lower bound of $\Omega(\sqrt{NT})$ with matching upperbounds is established, where N denotes the dimension of the set of arms. For the case when the number of arms is infinite or form a bounded subset of a N -dimensional space, a lower bound of $\Omega(N\sqrt{T})$ is established in [4, 1] with matching achievable bounds.

Several variants and special cases of stochastic linear bandits are available depending on what forms the set of arms. The classical stochastic multi-armed bandits introduced by Robbins [11] and later studied by Lai and Robbins [8] is a special case of linear bandits where the set of actions available in each round is the standard orthonormal basis. Auer [12] first studied stochastic linear bandits as an extension of “associated reinforcement learning” introduced in [13]. Since then several variants of the problems have been studied motivated by various applications. In [2, 14], the linear bandit setting is adopted to study content-based recommendation systems where the set of actions can change at each round (contextual), but their number is fixed. Another variant of linear bandits with finite action set are *spectral bandits* [15, 16], where the graph structure defines the set of actions and its size. Several authors [4, 1, 17] have considered linear bandits with arms constituting a (bounded) subset of a finite-dimensional vector space and remains fixed over the learning period. [18] considers cases where the set of arms can change between the rounds but must belong to a bounded subset of a fixed finite-dimensional vector space.

The paper is organized as follows: In Section 2, we describe the problem and setup notations. In Section 3, we derive a lower bound on expected regret and describe our main algorithm SEE and its variant SEE2. In Section 5, we analyze the performance of SEE, and its adaptation for general polyhedron is discussed in Section 6. In Section 7 we provide probability 1 bounds on the regret of SEE. Finally, we numerically compare performance of our algorithm against state-of-the-art in 8.

2 Problem formulation

We consider a stochastic linear optimization problem with bandit feedback over a set of arms defined by a polyhedron. Let $\mathcal{C} \subset \mathcal{R}^N$ denote a bounded polyhedral set of arms given by

$$\mathcal{C} = \{\mathbf{x} \in \mathcal{R}^N : \mathbf{A}\mathbf{x} \leq \mathbf{b}\} \quad (1)$$

where $\mathbf{A} \in \mathcal{R}^{M \times N}$, $\mathbf{b} \in \mathcal{R}^M$. At each round t , selecting an arm $x_t \in \mathcal{C}$ results in reward $r_t(\mathbf{x}_t)$. We investigate the case where the expected reward for each arm is a linear function regardless of the history. I.e., for any history \mathcal{H}_t , there is a parameter $\boldsymbol{\theta} \in [-1, 1]^N$, fixed but unknown, such that

$$\mathbb{E}[r_t(\mathbf{x})|\mathcal{H}_t] = \boldsymbol{\theta}'\mathbf{x} \quad \text{for all } t \text{ and } \mathbf{x} \in \mathcal{C}.$$

Under these setting the noise sequence $\{\nu_t\}_{t=1}^\infty$, where $\nu_t = r_t(\mathbf{x}) - \mathbf{x}'\boldsymbol{\theta}$ forms a martingale difference sequence. Let $\mathcal{F}_t = \sigma\{\nu_1, \nu_2, \dots, \nu_t, \mathbf{x}_1, \dots, \mathbf{x}_{t+1}\}$ denote the σ -algebra generated by noise events and arms selections till time t . Then ν_t is \mathcal{F}_t -measurable and we assume that it satisfies

$$\text{for all } b \in \mathcal{R}^1 \quad \mathbb{E}[e^{b\nu_t}|\mathcal{F}_{t-1}] \leq \exp\{b^2 R^2/2\}, \quad (2)$$

i.e., noise is conditionally R -sub-Gaussian which automatically implies $\mathbb{E}[\nu_t|\mathcal{F}_t] = 0$ and $\text{Var}(\nu_t) \leq R^2$. We can think of R^2 as the conditional variance of noise. An example of R -sub-Gaussian noise is $\mathcal{N}(0, R^2)$, or any bounded distribution over an interval of length $2R$ and zero mean. In our work, R is fixed but unknown.

A policy $\phi := (\phi_1, \phi_2, \dots)$ is a sequence of functions $\phi_t : \mathcal{H}_{t-1} \rightarrow \mathcal{C}$ such that an arm is selected in round t based on the history \mathcal{H}_{t-1} . Define expected (pseudo) regret of policy ϕ over T -rounds as:

$$R_T(\phi) = T\boldsymbol{\theta}'\mathbf{x}^* - \mathbb{E}\left[\sum_{t=1}^T \boldsymbol{\theta}'\phi(t)\right] \quad (3)$$

where $\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{C}} \boldsymbol{\theta}'\mathbf{x}$ denotes the optimal arm in \mathcal{C} , which exists and is an extremal point¹ of the polyhedron \mathcal{C} [19]. The expectation is over the random realization of the arm selections induced

¹Extremal point of a set is a point that is not a proper convex combination of points in the set.

by the noise process. The goal is to learn a policy that keeps the regret as small as possible. We will be also interested in regret of the policy defined as

$$\bar{R}_T(\phi) = T\theta'x^* - \sum_{t=1}^T \theta' \phi(t). \quad (4)$$

For the above setting, we can use *ConfidenceBall*₂ [4] or *UncertaintyEllipsoid* [1] and achieve optimal regret of order $N\sqrt{T}$. For linear bandits over a set with finite number of extremal points, one can also achieve regret that scales more gracefully, growing logarithmically in time T , using algorithms for the standard multi-armed bandits. Indeed, from fundamentals of linear programming

$$\arg \max_{\mathbf{x} \in \mathcal{C}} \theta' \mathbf{x} = \arg \max_{\mathbf{x} \in \mathcal{E}(\mathcal{C})} \theta' \mathbf{x},$$

where $\mathcal{E} := \mathcal{E}(\mathcal{C})$ denotes the set of extremal points of \mathcal{C} . Since the set of extremal points is finite for a polyhedron, we can use the standard Lai and Robbin's algorithm [8] or UCB1 in [6] treating each extremal point as an arm and obtain regret bound (problem dependent) of order $\frac{|\mathcal{E}|}{\Delta} \log T$, where $\Delta := \theta'x^* - \max_{\mathbf{x} \in \mathcal{E} \setminus \{x^*\}} \theta' \mathbf{x}$ denotes the gap between the best and the next best extremal point. However, the leading term in these bounds can be exponential in N , rendering these algorithm ineffective. For example, the number of extremal points of \mathcal{C} can be of the order $\binom{M+N}{M} = \mathcal{O}((2N)^M)$. Nevertheless, in analogy with the problem independent regret bounds in linear bandits, one wishes to derive problem dependent logarithmic regret where the dependence on set of arms is only linear in its dimension. Hence we seek an algorithm with regret of order $N \log T$.

In the following, we first derive a lower bound on the expected regret and develop an algorithm that is (almost) asymptotically optimal.

3 Main results

In this section we provide a lower bound on the expected regret and present our proposed policy and prove the main results regarding its complexity.

3.1 Lower Bound

We establish through a simple example that regret of any asymptotically optimal linear bandit algorithm is lower bounded as $\Omega(N \log T)$. Recall the fundamental property of the linear optimization that an optimal point is always an extremal point. Then any linear bandit algorithm on a polyhedral set of arms always play the extremal points. We exploit this fact, and mapping the problem to a standard multi-armed bandits we obtain the lower bound.

We need the following notations to prove the result. Let $\{\eta(\beta)\}_{\beta \in [0,1]}$ denote a set of distributions parametrized by $\beta \in [0, 1]$ and such that each $\eta(\beta)$ is absolutely continuous with respect to a positive measure m on \mathcal{R} . Let $p(x; \beta)$ denote the probability density function associated with distribution $\eta(\beta)$, and let $KL(\beta_1, \beta_2)$ denote the Kullback-Leibler (KL) divergence between distributions $\eta(\beta_1)$ and $\eta(\beta_2)$ defined as $KL(\beta_1, \beta_2) = \int_{\mathcal{R}} p(x; \beta_1) \log \frac{p(x; \beta_1)}{p(x; \beta_2)} m(dx)$. Consider a set of K arms. We say that arm k is parametrized by β_k if its reward is distributed according to $\eta(\beta_k)$.

We are now ready to state asymptotic lower bound for the linear bandit problem over any bounded polyhedron with positive measure. Without loss of generality, we restrict our attention to uniformly good policies as defined in [8]. We say that a policy ϕ is uniformly optimal if for all $\theta \in \Theta$, $R(T, \phi) = o(T^\alpha)$ for all $\alpha > 0$.

Theorem 1 *Let ϕ any uniformly good policy on a bounded polyhedron with positive measure. For any $\theta \in [0, 1]^N$, let $\mathbb{E}[\eta(\theta_k)] = \theta_k$ for all $k = 1, 2, \dots, N$. Then,*

$$\liminf_{T \rightarrow \infty} \frac{R_T(\phi)}{\log T} \geq \frac{(N-1)\Delta}{\max_{k: \theta_k < \theta^*} KL(\theta^*, \theta_k)} \quad \text{where} \quad \theta^* = \arg \max_n \theta_n \quad (5)$$

Proof sketch: First, note that number of extremal points of any bounded polyhedron with positive measure is atleast $(N+1)$. We can then restrict to a bounded polyhedron with $N+1$ extremal

points. Let $\tilde{\mathcal{C}} = \{\mathbf{x} \in \mathcal{R}^N : 0 \leq x_i \leq 1 \forall i = 1, 2, \dots, N\}$. The $(N + 1)$ extremal points of $\tilde{\mathcal{C}}$ are $\{\mathbf{e}_n : n = 1, 2, \dots, N\} \cup \{0\}$. In the linear bandit problem with unknown parameter θ , playing the extremal point \mathbf{e}_n gives mean reward θ_n . Also, by the property of linear optimization, any OFU policy will only play extremal points in every round. Then, the linear bandit over polyhedron $\tilde{\mathcal{C}}$ is the same as $N + 1$ -armed bandit where reward of k th arm $k = 1, 2, \dots, N$ is distributed as $\eta(\theta_k)$ with mean θ_k , and the reward of $N + 1$ th arm is distributed as $\eta(0)$ with mean 0.

The result follows from Lai-Robbin's lower bound for stochastic multi-armed bandits proved in [8] after verifying that the mean values of the parametrized distribution satisfy the required conditions.

3.2 Algorithms

The basic idea underlying our proposed technique is based on the following observations for linear optimization over a polyhedron. 1) The set of extremal points of polyhedron is finite and hence $\Delta > 0$. 2) When $\hat{\theta}$ is sufficiently close to θ , then over the set \mathcal{C} both $\arg \max \theta' \mathbf{x}$ and $\arg \max \hat{\theta}' \mathbf{x}$ give the same value. We exploit these observations and propose a two stage technique, where we first estimate θ based on a block of samples and then exploit it for much longer block. This is repeated with increasing block lengths so that at each point the regret is logarithmic. For ease of exposition, we first consider the polyhedron that contains origin and postpone the general case to Section 6.

Assume that the polyhedron $\mathcal{C} = \{\mathbf{x} \in \mathcal{R}^N : \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ contains origin as an interior point. Let \mathbf{e}_n denote n th standard unit vector of dimension N . For all $1 \leq n \leq N$, let $\bar{z}_n = \max \{z \geq 0, z\mathbf{e}_n \in \mathcal{C}\}$. The subset of arms $B := \{\bar{z}_n \mathbf{e}_n : n = 1, 2, \dots, N\}$ are the vertices of the largest simplex bounded in \mathcal{C} . Since $\theta_n = \theta' \mathbf{e}_n$ we can estimate θ_n by repeatedly playing the arm $\bar{z}_n \mathbf{e}_n$. One can also estimate θ_n by playing an interior point $z\mathbf{e}_n \in \mathcal{C}$ for some $z > 0$. But as will see later selecting the maximum possible z improves the probability of estimation error.

Algorithm-SEE

In our policy- which we refer as Sequential-Estimation-Exploitation (SEE)- we split the time horizon into cycles and each cycle consists of an exploration interval followed by an exploitation interval. We index the cycles by c and denote the exploration and exploitation intervals in cycle c as E_c and R_c , respectively. In the exploration interval E_c , we play each arm in B repeatedly for $(2c + 1)$ times. At the end of E_c , using the rewards observed for each arm in B in the past c - cycles we compute ordinary least square (OLS) to estimate each component $\theta_n, n = 1, 2, \dots, N$ separately and obtain the estimate $\hat{\theta}(c)$. Using $\hat{\theta}(c)$ as a proxy for θ , we compute a greedy arm $\mathbf{x}(c)$ by solving a linear program and play it repeatedly for $2^{c^2/(1+\epsilon)}$ times in the exploitation interval R_c , where $\epsilon > 0$ in an input parameter. We repeat the process for each cycle. A formal description of SEE is given in the adjacent figure. The estimation in line 13 is computed for all $n = 1, 2, \dots, N$ as follows:

$$\hat{\theta}_n(c) = \frac{1}{c^2} \sum_{i=0}^c \sum_{j=1}^{2i+1} r_{t_{i,n,j}} / z_n, \quad (6)$$

Algorithm 1 SEE

```

1: Input:
2:  $\mathcal{C}$ : The polyhedron
3:  $\epsilon$ : Algorithm parameter
4: Initialization:
5: Compute the set  $B$ 
6: for  $c = 0, 1, 2, \dots$  do
7:   Exploration:
8:   for  $n = 1 \rightarrow N$  do
9:     for  $j = 1 \rightarrow 2c + 1$  do
10:      Play arm  $z_n \mathbf{e}_n \in B$ ,
      observe reward  $r_{t_{c,n,j}}$ 
11:   end for
12:   Compute  $\hat{\theta}_n(c)$ 
13: end for
14:  $\hat{\theta}(c) \leftarrow (\hat{\theta}_1(c), \hat{\theta}_2(c), \dots, \hat{\theta}_N(c))$ 
15:  $\mathbf{x}(c) \leftarrow \arg \max_{\mathbf{x} \in \mathcal{C}} \mathbf{x}' \hat{\theta}(c)$ 
16: Exploitation:
17: for  $j = 1 \rightarrow \lfloor 2^{c^2/(1+\epsilon)} \rfloor$  do
18:   Play arm  $\mathbf{x}(c)$ , observe reward
19: end for
20: end for

```

Note that in the exploration intervals, SEE plays a fixed set of arms and no adaption happens, adding positive regret in each cycle. The regret incurred in the exploitation intervals starts reducing as the estimation error gets small, and when it falls below $\Delta/2$ the step (line-16) selects the optimal arm and no regret is incurred in the exploitation intervals (Lemma 2). As we will show later, the probability of estimation error decays super-exponentially across the cycles, and hence the probability of playing a sub-optimal arm in the exploitation interval also decays super-exponentially.

Theorem 2 *Let the noise be R -sub-Gaussian and without loss of generality² assume $\theta \in [-1, 1]^N$. Then, the expected regret of SEE, with parameter $\epsilon > 0$ is bounded as follows:*

$$R_T(SEE) \leq 2R_m N \log^{1+\epsilon} T + 4R_m N \gamma_1, \quad (7)$$

where R_m denotes the maximum reward. γ_1 is a constant that depends on noise parameter R and the sub-optimality gap Δ .

The ϵ parameter determines the length of the exploitation intervals, and larger ϵ implies that SEE spends less time in exploitation and more time in exploration. Increasing ϵ will make SEE spend more time in explorations resulting in improved estimations and reduces the probability of playing sub-optimal arm in the exploitation intervals. Hence parameter ϵ determines how fast the regret concentrates, and larger its value more 'risk-averse' is the algorithm. This motivates us to consider a variant of SEE that is more risk averse but at the cost of increased expected regret.

3.3 Risk Averse Variant

Our second algorithm-which we refer to as SEE2- is essentially same as SEE, except for the length of the exploration intervals which is exponential instead of super-exponential and does not depend on ϵ . Specifically, we play the greedy arm 2^c times in cycle c . Compared to SEE, SEE2 spends significantly more time in the exploration intervals, and hence the probability that it makes error in the exploitation intervals is also significantly smaller and thus its regret concentrates around the expected regret faster.

Theorem 3 *Let the noise be R -sub-Gaussian and $\theta \in [-1, 1]^N$. Then, the expected regret of SEE2 is bounded as follows:*

$$R_T(SEE2) \leq 2R_m N \log^2 T + 4N R_m \gamma_2 \quad (8)$$

where γ_2 is a constant that depends on noise parameter R and the sub-optimality gap Δ .

4 Optimal Algorithm.

We next obtain an optimal algorithm that achieves the lower bound in (5) within a constant factor when the sub-Gaussian parameter R is known.

²For general θ , we replace it by $\frac{\theta}{\|\theta\|_\infty}$ and the same method works. Only R_m is scaled by a constant factor.

Algorithm-PolyLin:

In our next policy- which we refer as **Polyhedral-Linear-bandits** we again split the time horizon into cycles consisting of an exploration interval followed by an exploitation interval as in SEE. As earlier, we index the cycles by i and denote the exploration and exploitation intervals in cycle i as E_i and R_i , respectively. In the exploration interval E_i , we play each arm in \mathcal{B} once. After c -cycles, using the rewards observed for each arm in \mathcal{B} in the past $\{E_i, i = 1, 2, \dots, c\}$ exploration intervals we compute ordinary least square (OLS) to estimate each component $\theta_n, n = 1, 2, \dots, N$ separately, and obtain the estimate $\hat{\theta}(c)$ as follows.

$$\hat{\theta}_n(c) = \frac{1}{c} \sum_{i=1}^c r_{t_{i,n}} / z_n, \quad (9)$$

Using $\hat{\theta}(c)$ as a proxy for θ we compute a greedy arm $\mathbf{x}(c)$ and the sub-optimality gap $\hat{\Delta}(c)$ as follows.

$$\hat{\Delta}(c) = \mathbf{x}'(c)\theta(c) - \max_{\mathbf{x} \in \mathcal{C} \setminus \mathbf{x}(c)} \mathbf{x}'\theta(c).$$

In the exploitation interval R_c , we play $\mathbf{x}(c)$ repeatedly for $2^{\kappa(c)c}$ times where $\kappa(c)$ is set to $a\hat{\Delta}(c)/2$, where $a = \min_n \bar{z}_n / R^2$. We repeat the process for each cycle. A formal description of PolyLin is given in the adjacent figure.

Algorithm 2 PolyLin

```

1: Input:
2:  $\mathcal{C}$ : The polyhedron
3:  $R$ : Noise parameter
4: Initialization
5: Compute the set  $\mathcal{B}$ 
6:  $a := \min_n \bar{z}_n^2 / R^2$ 
7: for  $i = 1, 2, \dots$  do
8:   Exploration:
9:   for  $n = 1 \rightarrow N$  do
10:     Play arm  $z_n \mathbf{e}_n \in \mathcal{B}$ 
11:     observe reward  $r_{t_{i,n}}$ 
12:    $c = i$ , Compute  $\hat{\theta}_n(c)$  as in (9)
13:   end for
14:    $\hat{\theta}(c) \leftarrow (\hat{\theta}_1(c), \hat{\theta}_2(c), \dots, \hat{\theta}_N(c))$ 
15:    $\mathbf{x}(c) \leftarrow \arg \max_{\mathbf{x} \in \mathcal{C}} \mathbf{x}'\hat{\theta}(c)$ 
16:    $\kappa(c) \leftarrow a\hat{\Delta}(c)/2$ 
17:   Exploitation:
18:   for  $j = 1 \rightarrow \lfloor 2^{\kappa(c)c} \rfloor$  do
19:     Play arm  $\mathbf{x}(c)$ , observe reward
20:   end for

```

Note that the exploration intervals of PolyLin are fixed length, whereas in SEE they are increasing as the time progresses. Also, exploitation intervals in PolyLin are adaptive, whereas it is non-adaptive in SEE.

Theorem 4 *Let the noise be R -sub-Gaussian and without loss of generality assume $\theta \in [-1, 1]^N$. Then, the expected regret of PloyLin is bounded as follows:*

$$R_T(\text{PolyLin}) \leq 2R_m N \frac{\log T}{\kappa} + 4R_m N \gamma_3, \quad (10)$$

where R_m denotes the maximum reward. γ_3 and κ are constants that depends on noise parameter R and the sub-optimality gap Δ .

5 Regret Analysis

In this section we prove Theorem 2, the proof of Theorem 3 follows similarly and omitted. We first derive the probability of error in estimating each component of θ in each cycle. Note that in the exploration stage of each cycle c we sample each arm $z_n \mathbf{e}_n \in \mathcal{B}, i = 1, 2, \dots, N$, 2 times more than that in the exploration stage of the previous cycle. Thus, we have c^2 plays of each arm $z_n \mathbf{e}_n \in \mathcal{B}$ at the end of cycle c . The estimation error of component θ_n after c -cycles is given as follows:

Lemma 1 *Let the noise be R -sub-Gaussian and $\delta > 0$. In any cycle c of both SEE and SEE2, for all $n = 1, 2, \dots, N$ we have*

$$P\left(\left|\hat{\theta}_n(c) - \theta_n\right| > \delta\right) \leq 2 \exp\{-c^2 \delta^2 \bar{z}_n^2 / 2R^2\}. \quad (11)$$

Note that larger the value of \bar{z}_n , the smaller the probability of estimation error is. The next lemma gives the probability that we play a suboptimal arm in the exploitation intervals of a cycle.

Lemma 2 For every cycle c , we have

a. Let $a := \min_n \bar{z}_n^2 / R^2$. The estimation error is bounded as

$$\Pr\{\|\hat{\theta}(c) - \theta\|_\infty > \eta\} \leq 2N \exp\{-c^2 \eta^2 a\}, \quad (12)$$

b. Let $h = \sup_{\mathbf{x} \in \mathcal{C}} \|\mathbf{x}\|_1$. The error in reward estimation is bounded as

$$\Pr\left(\exists \mathbf{x} \in \mathcal{C} \text{ such that } \left|\hat{\theta}'(c)\mathbf{x} - \theta'\mathbf{x}\right| > \eta\right) \leq 2Ne^{-\frac{c^2 \eta^2 a}{h^2}}. \quad (13)$$

c. Probability that we play a sub-optimal arm is bounded as

$$\Pr\left(\arg \max_{\mathbf{x} \in \mathcal{C}} \hat{\theta}(c)'\mathbf{x} \neq \arg \max_{\mathbf{x} \in \mathcal{C}} \theta'\mathbf{x}\right) \leq 2Ne^{-\frac{ac^2 \Delta^2 / 4}{h^2}}. \quad (14)$$

The proofs of Lemmas 1 and 2 are given in appendix. Recall that the number of extremal points is finite for the polyhedron \mathcal{C} and $\Delta > 0$. We use this fact to argue that whenever $\|\hat{\theta}(c) - \theta\|_\infty < \Delta/2$, the greedy stage of the algorithm selects the optimal arm. This is an importation observation and follows from continuity property of optimal point in linear optimization theory [19]. Further, the probability of this event decays super-exponentially fast in our policy implying that the probability that we incur a positive regret in the exploitation intervals is gets negligibly small over the cycles. We compute the expected regret incurred in the exploration and exploitation intervals separately.

5.1 Regret of SEE.

We analyze the regret in the Exploration and Exploitation phases separately as follows.

Exploration regret: At the end of cycle c , each arm in \mathcal{B} is played $\sum_{i=1}^c (2i+1) = c^2$ times. The total expected regret from the exploration intervals after c cycles is at most $Nc^2 R_m$.

Exploitation regret: Total expected regret from the exploration intervals after c cycle is

$$4NR_m \sum_{i=1}^c 2^{i^{2/(1+\epsilon)}} 2^{-i^2 a \Delta^2} = 4NR_m \sum_{i=1}^c 2^{i^{2/(1+\epsilon)} - i^2 a \Delta^2} \leq 4NR_m \gamma_2 \quad (15)$$

where $\gamma_2 := \sum_{i=1}^\infty 2^{i(i^{(1-\epsilon)/(1+\epsilon)} - c_1 i \Delta^2 / 4)}$ is a convergent series. After c cycles, the total number of plays is $T = \sum_{i=1}^c e^{i^{\frac{2}{1+\epsilon}}} + Nc^2 \geq e^{c^{\frac{2}{1+\epsilon}}}$ and we get $c^2 \leq \log^{1+\epsilon} T$. Finally, expected regret from T -rounds is bounded as

$$R_T(SEE) \leq 2R_m N \log^{1+\epsilon} T + 4NR_m \gamma_2 = \mathcal{O}(N \log^{1+\epsilon} T).$$

5.2 Regret of PolyLin.

We analyze the regret in the Exploration and Exploitation phases separately as follows.

Exploration regret: After c cycles, each arm in \mathcal{B} is played c times. The total expected regret from the exploration intervals after c cycles is at most NcR_m .

Exploitation regret: Total expected regret from the explorations interval after c cycles is

$$4NR_m \sum_{i=1}^c 2^{\kappa(i)} 2^{-ia \Delta^2} = 4NR_m \sum_{i=1}^c 2^{i\kappa(i) - ia \Delta^2} \leq 4NR_m \sum_{i=1}^\infty 2^{ia\{\hat{\Delta}^2(i)/2 - \Delta^2\}}. \quad (16)$$

Now consider the series $\gamma_3 := \sum_{i=1}^\infty 2^{ia\{\hat{\Delta}^2(i)/2 - \Delta^2\}}$.

- From Lemma 2(a), $\theta(c) \rightarrow \theta$ as $c \rightarrow \infty$ almost surely, we get $\hat{\mathbf{x}}(c) \rightarrow \mathbf{x}^*$ almost surely and which in turn implies $\hat{\Delta}(c) \rightarrow \Delta$ almost surely.
- Then, for $0 < \epsilon < \Delta^2/4$, the difference $\hat{\Delta}(c)^2/2 - \Delta^2 \leq -\Delta^2/2 + \epsilon < 0$ for all but finitely many c . Hence, γ_3 is finite.

After c cycles the total number of plays is $T = \sum_{i=1}^c 2^{i\kappa(i)} + Nc \geq 2^{c\kappa(c)}$, and we get $c \leq \frac{\log T}{\kappa(c)}$. Finally, expected regret form T -rounds, as $T \rightarrow \infty$, is bounded as

$$R_T(\text{PolyLin}) \leq 2R_m N \frac{\log T}{\kappa(c)} + 4NR_m\gamma_3.$$

Note that $\hat{\Delta}(c)^2/2 - \Delta^2 \geq -\Delta^2/2 - \epsilon$ for all but finitely many c . Then for sufficiently large c we get $k(c)/a \geq \Delta^2/2 - \epsilon \geq \Delta^2/4$. Substituting in the last inequality we get

$$R_T(\text{PolyLin}) \leq 8R_m N \frac{\log T}{a\Delta} + 4NR_m\gamma_3 = \mathcal{O}(N \log T).$$

6 General Polyhedron

In this section we extend the analysis of the previous section to the case where origin is not an interior point of \mathcal{C} .

Analogous to set \mathcal{B} , we first define a set of arms that lie on the boundary of the polyhedron and these points are computed with respect to an interior point $\bar{\mathbf{x}}$ of \mathcal{C} that we use as a proxy for origin. We use OPT-1 to find an interior point, whose smallest distance to boundaries along all the directions $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_N\}$ is the largest. The motivation to maximize the minimal distances to the boundaries comes from lemma 2, where larger value of a imply smaller probability of estimation error.

OPT-1:

$$(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = \arg \max_{\mathbf{x}} \min_i y_i$$

subjected to:

$$\mathbf{A}\mathbf{x} \leq \mathbf{b}$$

$$y_i \geq 0 \quad \forall i = 1, 2, \dots, N$$

$$\mathbf{A}(\mathbf{x} + y_i \mathbf{e}_i) \leq \mathbf{b} \quad \forall i = 1, 2, \dots, N$$

$$\mathbf{A}(\mathbf{x} - y_i \mathbf{e}_i) \leq \mathbf{b} \quad \forall i = 1, 2, \dots, N$$

OPT-2:

$$(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = \arg \max_{\mathbf{x}, \mathbf{y}, \alpha} \alpha$$

subjected to:

$$\alpha > 0; \quad \mathbf{A}\mathbf{x} \leq \mathbf{b}$$

$$y_i - \alpha > 0 \quad \forall i = 1, 2, \dots, N$$

$$\mathbf{A}(\mathbf{x} + y_i \mathbf{e}_i) \leq \mathbf{b} \quad \forall i = 1, 2, \dots, N$$

$$\mathbf{A}(\mathbf{x} - y_i \mathbf{e}_i) \leq \mathbf{b} \quad \forall i = 1, 2, \dots, N$$

OPT-1 can be translated into an equivalent linear programme given in OPT-2 and hence the point $\bar{\mathbf{x}}$ can be efficiently computed. We note that the set of points $\{\bar{\mathbf{x}} + y_n \mathbf{e}_n : n = 1, 2, \dots, N\}$ need not all necessarily lie on the boundary. To see this, let the point $\bar{\mathbf{x}}$ returned by OPT-1 is such that it is closer to the boundary along i th direction. Then the vector with all its component equal to y_i is a solution of OPT-1. To overcome this, we further stretch each point $\bar{\mathbf{x}} + y_n \mathbf{e}_n$ along the direction \mathbf{e}_n such that it hits the boundary. Let $\bar{\mathbf{z}}_n = \arg \max_z \{ |z| : z \mathbf{e}_n \in \mathcal{C} \}$. Finally, we fix the set of arms we use for explorations as $\bar{\mathcal{B}} = \{\bar{\mathbf{z}}_n \mathbf{e}_n + \bar{\mathbf{x}} : n = 1, 2, \dots, N\}$.

We are now ready to present an algorithm for linear bandits over for any polyhedra. For the general polyhedron, we use the SEE with the exploration strategy modified as follows. In cycle c , we first play the arm $\bar{\mathbf{x}}$ for $2c + 1$ and then play each arm in $\bar{\mathcal{B}}$ $2c + 1$ times as earlier. To estimate the component θ_n , we average the difference in rewards observed from arms $\bar{\mathbf{x}} + \bar{\mathbf{z}}_n \mathbf{e}_n$ and $\bar{\mathbf{x}}$ so far. From a straightforward modification of regret analysis of SEE, we can show that the expected regret of modified algorithm is upper bounded as $\mathcal{O}(N \log^{1+\epsilon} T)$ for all $\epsilon > 0$.

The new algorithm required that we play the arm $\bar{\mathbf{x}}$ along with the arms in $\bar{\mathcal{B}}$ in the exploration intervals to obtain estimate of $\boldsymbol{\theta}$, and it increases the length of exploration intervals. However, it is possible that one can obtain estimates only by playing arms in $\bar{\mathcal{B}}$ provided we suitably modify the estimation method. More details are given in the appendix.

7 Probability 1 Regret Bounds

Recall the definition of expected regret and regret in (3) and (4). In this section we show that with probability 1, the regret of our algorithms are within a constant factor from their expected regret.

Theorem 5 *With probability 1, $\bar{R}_T(\text{SEE})$ is $\mathcal{O}(N \log^{1+\epsilon} T)$ and $\bar{R}_T(\text{SEE2})$ is $\mathcal{O}(N \log^2 T)$.*

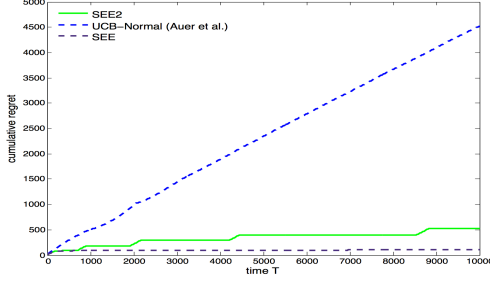


Figure 1: Regret comparison against multi-armed bandits, arms are corners of 10-dim. hypercube.

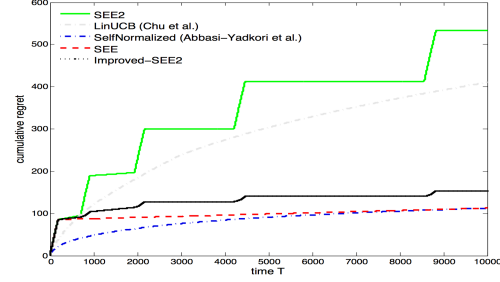


Figure 2: Regret comparison against linear bandit algorithms on 10-dim. hypercube.

Proof: Let \mathbb{C}_n denote an event that we select sub-optimal arm in the n th cycle. From Lemma 2, this event is bounded as $\Pr\{\mathbb{C}_n\} \leq N \exp\{-\mathcal{O}(n^2)\}$. Hence $\sum_{n=1}^{\infty} \Pr\{\mathbb{C}_n\} < \infty$. Now, from application of Borel-Cantelli lemma, we get $\Pr\{\limsup_{n \rightarrow \infty} \mathbb{C}_n\} = 0$, which implies that almost surely SEE and SEE2 play optimal arm in all but finitely many cycles. Hence the exploitation intervals contribute only a bounded regret. Since the regret due to exploration intervals is deterministic, the regret of SEE and SEE2 are within a constant factor from their expected regret with probability 1, i.e., $\Pr\{\exists C_1 \text{ such that } \bar{R}_T(\text{SEE}) \leq R_T(\text{SEE}) + C_1\}$ and $\Pr\{\exists C_2 \text{ such that } \bar{R}_T(\text{SEE2}) \leq R_T(\text{SEE2}) + C_2\}$. This completes the claim.

We note that the regret bounds proved in [4] hold with high confidence, where as ours hold with probability 1 and hence provides a stronger performance guarantee.

8 Experiments

In this section we investigate numerical performance of our algorithms against the known algorithms. We run the algorithms on a hypercube with dimension $N = 10$. We generated $\theta \in [0, 1]^N$ randomly and noise is zero mean Gaussian random variable with variance 1 in each round. The experiments are averaged over 10 runs. In Fig. 1 we compare SEE ($\epsilon = 0.3$) and SEE2 against UCB-Normal [20], where we treated each extremal point as an arm of an 2^N -armed bandit problem. As expected, our algorithms perform much better. UCB-Normal need to sample each of the 2^N atleast once before it could start learning the right arm. Whereas, our algorithm starts playing the right arm after a few cycles of exploration intervals. In Fig. 2, we compare our algorithms against the linear bandits algorithm LinUCB and self-normalization based algorithm in [18], which is labeled SelfNormalized in the figure. For these we set confidence parameter to 0.001. We see that SEE beats LinUCB by a huge margin, but its performance comes close to that of SelfNormalized algorithm. Note that SelfNormalized algorithm requires knowledge of sub-Gaussianity parameter R of noises super. Whereas, our algorithms are agnostic to this parameter. Though, SEE2 seems to play the right arm in exploitation intervals, its regret performance is poor. This is due to increased number of exploration intervals, where no adaptation happens and a positive regret is always incurred.

The numerical performance of SEE2 can be improved by adaptively playing the arms in the exploration plays as follows, but at the increase cost of computations complexity. In each cycle $c + 1$, we find a new set \bar{B} computed by setting \bar{x} to $\mathbf{x}(c)$, the greedy arm selected in the previous cycle, and play the new set arms as in the explorations intervals of the algorithm given for the general polyhedron. However, since $\mathbf{x}(c)$ is an extremal points some of the \bar{z}_n 's are zero. To overcome this, we slightly shift the point $\mathbf{x}(c)$ into the interior of the polyhedron along the direction $\mathbf{x}(c) - \bar{x}$ and find a new set \bar{B} with respect to the new interior point. The regret of the algorithm based on this adaptive exploitation strategy is shown in Fig. 2 with label 'Improved-SEE2'. As shown, the modification improves performance of SEE2 significantly. In all the numerical plots, we initialized the algorithm to run from cycle number 5.

9 Conclusion

We studied stochastic linear optimization over polyhedral set of arms with bandit feedback. We provided asymptotic lower bound for any policy and developed algorithms that are near asymptotically optimal. The regret of the algorithms grow (near) logarithmically in T and its growth rate is linear

in the dimension of the polyhedron. We showed that the regret upper bounds hold almost surely. The regret growth rate of our algorithms is $\log^{1+\epsilon} T$ for some $\epsilon > 0$. It is interesting to develop strategies that work for $\epsilon = 0$, while still maintain linear growth rate in N .

References

- [1] P. Rusmevichientong and J. N. Tsitsiklis, “Linearly parameterized bandits,” *Mathematics of Operations Research*, vol. 35, no. 2, pp. 395–411, 2010.
- [2] L. Li, C. Wei, J. Langford, and R. E. Schapire, “A contextual-bandit approach to personalized news article recommendation,” in *Proceeding of International Word Wide Web conference, WWW*, NC, USA, April 2010.
- [3] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*. Cambridge University Press, 2010.
- [4] V. Dani, T. P. Hayes, and S. M. Kakade, “Stochastic linear optimization under bandit feedback,” in *Proceeding of Conference on Learning Theory, COLT*, Helsinki, Finland, July 2008.
- [5] J.-Y. Audibert, R. Munos, and C. Szepesvári, “Exploration-exploitation tradeoff using variance estimates in multi-armed bandits,” *Theoretical Computer Science*, vol. 410, p. 18761902, 2009.
- [6] P. Auer, Nichol6-Cesa-Bianchi, and P. Fischer, “Finite-time analysis of multiarmed bandit problem trade-offs,” *Journal of Machine Learning*, vol. 3, pp. 235–256, 2002.
- [7] J.-Y. Audibert and S. Bubeck, “Regret bounds and minimax policies under partial monitoring,” *Journal of Machine Learning Research*, vol. 11, pp. 2635–2686, 2010.
- [8] T. L. Lai and H. Robbins, “Asymptotically efficient adaptive allocation rules,” *Journal of Advances in applied mathematics*, vol. 6, no. 1, pp. 4–22, 1985.
- [9] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning, and Games*. Cambridge University Press, New York, 2006.
- [10] P. Auer, N. Cesa-Bianchi, Y. F. Robert, and E. Schapire, “The non-stochastic multi-armed bandit problem,” *SIAM Journal on Computing*, vol. 32, 2003.
- [11] H. Robbins, “Some aspects of the sequential design of experiments,” *Bulletin of the American Mathematics Society*, vol. 58, pp. 527–535, 1952.
- [12] P. Auer, “Using confidence bounds for exploitation-exploration trade-offs,” *Journal of Machine Learning Research*, vol. 3, pp. 397–422, 2002.
- [13] N. Abe and P. M. Long, “Associative reinforcement learning using linear probabilistic concepts,” in *Proceeding of International Conference on Machine Learning (ICML)*, 1999.
- [14] W. Chu, L. Li, L. Reyzin, and R. E. Schapire, “Contextual bandits with linear payoff functions,” in *Proceeding of International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011, pp. 208–214.
- [15] M. Valko, R. Munos, B. Kveton, and T. Kocák, “Spectral bandits for smooth graph functions,” in *Proceeding of International Conference on Machine Learning (ICML)*, 2014.
- [16] M. Hanawal, V. Saligrama, M. Valko, and R. Munos, “Cheap bandits,” in *Proceeding of International Conference on Machine Learning (ICML)(to appear)*, 2015.
- [17] Y. Abbasi-Yadkori, A. Antos, and C. Szepesvári, “Forced-exploration based algorithms for playing in stochastic linear bandits,” in *Proceeding COLT workshop on On-line Learning with Limited Feedback*, 2009.
- [18] Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári, “Improved algorithms for linear stochastic bandits,” in *Proceeding of Advances in Neural Information Processing Systems (NIPS)*, 2011, pp. 2312–2320.
- [19] D. Bertsimas and J. N. Tsitsiklis, *Introduction to Linear Optimization*. Athena Scientific, Belmont, Massachusetts, 2008.
- [20] P. Auer, N. Cesa-Bianchi, and P. Fischer, “Finite-time analysis of the multiarmed bandit problem,” *Machine Learning*, vol. 47, no. 2, pp. 235–256, 2002.

Proof of Lemma 1

Let $\epsilon_{t_{i,n,j}}$ denote the noise in reward from playing $z_n \mathbf{e}_n$ in phase i for the j th time. We bound the estimation error as follows:

$$P\left(\left|\hat{\theta}_n(c) - \theta_n\right| > \delta\right) \quad (17)$$

$$= P\left(\left|\sum_{i=1}^{c^2} \epsilon_{t_{i,n,j}} / c^2 z_n\right| > \delta\right) \quad (18)$$

$$= P\left(s \left|\sum_{i=1}^{c^2} \epsilon_{t_{i,n,j}}\right| > s c^2 z_n \delta\right) \quad (19)$$

$$= P\left(\exp\left\{s \left|\sum_{i=1}^{c^2} \epsilon_{t_{i,n,j}}\right|\right\} > \exp\{s c^2 z_n \delta\}\right) \quad (20)$$

$$\leq 2P\left(\exp\left\{s \sum_{i=1}^{c^2} \epsilon_{t_{i,n,j}}\right\} > \exp\{s c^2 z_n \delta\}\right) \quad (21)$$

$$\leq 2\mathbb{E}\left[\exp\left\{s \sum_{i=1}^{c^2} \epsilon_{t_{i,n,j}}\right\}\right] \exp\{-s c^2 z_n \delta\} \quad (22)$$

$$\leq 2 \prod_{i=1}^{c^2} \mathbb{E}\left[\exp\{s \epsilon_{t_{i,n,j}}\} \mid \mathcal{F}_{t-1}\right] \exp\{-s c^2 z_n \delta\} \quad (23)$$

$$\leq 2 \prod_{i=1}^{c^2} \exp\{s^2 \beta^2 / 2\} \exp\{-s c^2 z_n \delta\} \quad (24)$$

$$= 2 \exp\{c^2 (s^2 \beta^2 / 2 - s z_n \delta)\}, \quad (25)$$

where (18) follows from estimation step given in (6). In (19) and (20) we exponentiated both sides within the probability functions after multiplying them by $s > 0$. (21) follows by applying union bound and using the symmetric property of the noise terms. In (22) we applied the Markov inequality. In (23) we applied conditional independence property of the noise. (24) follows by applying the definition of sub-Gaussian property.

Note that upper bound in (25) holds for all $s > 0$ and is minimized at $s^* = \frac{\delta z_n}{\beta^2} > 0$. Finally, the lemma by substituting s^* in (25).

Proof of Lemma 2

Part a:

We bound the estimation error as follows:

$$\Pr\left(\left\|\hat{\boldsymbol{\theta}}(c) - \boldsymbol{\theta}\right\|_{\infty} > \eta\right) \quad (26)$$

$$\leq \Pr\left(\exists n : \left|\hat{\theta}_n(c) - \theta_n\right| > \eta\right) \quad (27)$$

$$\leq \sum_{n=1}^N \Pr\left(\left|\hat{\theta}_n(c) - \theta_n\right| > \eta\right) \quad (28)$$

$$\leq 2N c_1 e^{-ac^2 \eta^2}. \quad (29)$$

In (28) we applied the union bound result and in (29) we applied (11).

Part b:

For all $\mathbf{x} \in \mathcal{C}$, we have

$$|\mathbf{x}'\boldsymbol{\theta}(c) - \mathbf{x}'\boldsymbol{\theta}| \leq \|\boldsymbol{\theta}(c) - \boldsymbol{\theta}\|_\infty \|\mathbf{x}\|_1. \quad (30)$$

Define events $\mathbb{A} = \{\exists \mathbf{x} \text{ such that } |\mathbf{x}'\boldsymbol{\theta}(c) - \mathbf{x}'\boldsymbol{\theta}| > \eta\}$ and $\mathbb{B} = \{\|\boldsymbol{\theta}(c) - \boldsymbol{\theta}\|_\infty h > \eta\}$. The last inequality implies $\Pr\{\mathbb{A}\} \leq \Pr\{\mathbb{B}\}$. The claim follows from part-a of the lemma.

Part c:

Suppose $\mathbf{y} \neq \mathbf{x}^*$, where \mathbf{x}^* is the optimal arm, such that $\boldsymbol{\theta}'(c)\mathbf{y} > \boldsymbol{\theta}'(c)\mathbf{x}^*$. Then, since $\boldsymbol{\theta}'\mathbf{x}^* - \boldsymbol{\theta}'\mathbf{y} \geq \Delta$ we must have that either $|\boldsymbol{\theta}'\mathbf{x}^* - \boldsymbol{\theta}'(c)\mathbf{x}^*| \geq \Delta/2$ or $|\boldsymbol{\theta}'(c)\mathbf{y} - \boldsymbol{\theta}'\mathbf{y}| \geq \Delta/2$, otherwise we cannot close the gap. Hence, if the greedy selection in cycle c is not \mathbf{x}^* , it implies that there exists a $\mathbf{x} \in \mathcal{C}$ such that $|\boldsymbol{\theta}'(c)\mathbf{x} - \boldsymbol{\theta}\mathbf{x}| > \Delta/2$. From part-b this probability is bounded as $2N \exp\{-ac^2\eta^2/h\}$, where $\eta = \Delta/2$. This completes the proof.

Estimation in the case general polyhedron

Let $\bar{\mathbf{x}}_i = \bar{\mathbf{x}} + \alpha_i \mathbf{e}_i$. Let $\hat{r}_i(c) := \frac{1}{c^2} \sum_{i=1}^c \sum_{j=1}^{2c+1} r_{t_{i,n,j}}$ denote the average of the reward obtained from arm $\bar{\mathbf{x}}_i$ till end of phase m . At the end of phase m , we estimate $\boldsymbol{\theta}$ as follows:

$$\hat{\boldsymbol{\theta}}(c) = (\mathbf{1}\bar{\mathbf{x}}' + \mathbf{D}(\boldsymbol{\alpha}))^{-1} \hat{\mathbf{r}}(c),$$

where $\boldsymbol{\alpha}$ denote the diagonal matrix with diagonal elements as α and $\hat{\mathbf{r}}(c)$ is the vector with i th component as $\hat{r}_i(c)$. By applying matrix inversion lemma we get

$$\hat{\boldsymbol{\theta}}(c) = \left(\mathbf{D}^{-1}(\boldsymbol{\alpha}) - \frac{\mathbf{D}^{-1}(\boldsymbol{\alpha})\mathbf{1}\bar{\mathbf{x}}'\mathbf{D}^{-1}(\boldsymbol{\alpha})}{\bar{\mathbf{x}}'\mathbf{D}^{-1}(\boldsymbol{\alpha})\mathbf{1}} \right)$$

After simplification, for each $i = 1, 2, \dots, N$ we have

$$\hat{\theta}_i(c) = \frac{1}{\alpha_i} \left(\hat{r}_i(c) - \frac{\sum_{j=1}^N (\bar{x}_j/\alpha_j) \hat{r}_j(c)}{\sum_{l=1}^N \bar{x}_l/\alpha_l} \right)$$

Substituting the reward from arm $\bar{\mathbf{x}}_i$, i.e.,

$$r_{\bar{\mathbf{x}}_i} = \bar{\mathbf{x}}'\boldsymbol{\theta} + \alpha_i \theta_i + \epsilon$$

and further simplifying we get

$$\hat{\theta}_i(c) = \frac{1}{\alpha_i} \left(\alpha_i \theta_i - \bar{\mathbf{x}}'\boldsymbol{\theta} + \sum_{j=1}^N \beta_j \hat{\epsilon}_j(c) \right)$$

where $\beta_j = \frac{\bar{x}_j}{\alpha_j}$ and $\hat{\epsilon}_j(m)$ is the noise average from playing arm $\bar{\mathbf{x}}_i$.